2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Algorithms for Leader Selection in Large Dynamical Networks: Noise-Free Leaders

Makan Fardad, Fu Lin, and Mihailo R. Jovanović

***Abstract*— We consider networks of single-integrator systems, where it is desired to optimally assign a predetermined number of systems to act as leaders. Performance is measured in terms of the $\mathcal{H}_2$ norm of the overall network, and the leaders are assumed to always follow their desired state trajectories. We demonstrate that, after applying a sequence of relaxations, the problem can be formulated as a semidefinite program and thus solved efficiently. We compare the results of our algorithms against others reported in the literature. Finally, we interpret the leader selection problem in terms of electrical networks and Kron reduction theory.**

***Index Terms*— Consensus, convex optimization, convex relaxation, effective resistance, Kron reduction, ℓ_1 minimization, leader selection, semidefinite programing, sparsity.**

I. Introduction

The problem of optimal leader selection for networks of dynamical systems was recently formulated in [1]. Reference [1] considers a framework in which the follower (i.e., non-leader) nodes are modeled as single-integrators, and to compute their control signal they are restricted to using only relative information between themselves and those nodes they communicate with. The leader nodes, on the other hand, are able to maintain the desired state at all times and are not corrupted by input disturbances. It is then desired to optimally assign a predetermined number of nodes to act as leaders in order to minimize a performance index. The leader selection problem arises in several emerging applications, including opinion dynamics in social networks and multi-agent formation control [1].

The leader selection problem is inherently difficult due to its combinatorial nature; choosing k leaders among n agents means that the performance objectives for $\binom{n}{k}$ possible leader combinations would have to be compared; such an exhaustive search is not feasible for large systems. Therefore it is common practice to relax the binary constraints (a node is either a leader or it is not) that arise in these problems, in order to obtain a more tractable formulation.

In this paper, using the formulation proposed in [1], we present two methods for efficiently selecting leaders. Our methods rely on a linear approximation of the objective function and a relaxation of the binary constraints. Method 1 is based on a greedy algorithm, where leaders are chosen one at a time in an iterative procedure. This algorithm is numerically efficient but suffers from the fact that, by design, in choosing the $(\kappa+1)$th leader it leaves the first κ leaders untouched; the optimal choice of leaders does not necessarily have this feature. Method 2 is based on the sparsity-promoting property of the ℓ_1 norm [2], [3]. This method does not suffer from the aforementioned shortcoming of a greedy algorithm and is observed to perform well on benchmark examples, often finding the globally optimal choice of leaders.

Our presentation is organized as follows. In Section II we state the optimal leader selection problem, as formulated in [1]. In Section III we use a linear approximation and relaxation of binary constraints to convexify the leader selection problem, and then proceed to develop two optimization algorithms. In Section IV we demonstrate the performance and efficiency of our algorithms using the example provided in [1] as a benchmark. In Section V we give an electrical network interpretation to the leader selection problem. We close with conclusions and directions for future work in Section VI.

II. Problem Statement

In this section we first state the leader selection problem as formulated in [1]. We then elaborate on how this formulation is obtained.

We consider the optimal leader selection problem [1],

$$\begin{aligned} \text{minimize} \quad & \operatorname{trace}(L_h^{-1}) \\ \text{subject to} \quad & h_j \in \{0,1\}, \quad j \in 1,\dots,n \\ & \|h\|_1 = k \end{aligned} \tag{LS}$$

The vector h is composed only of zeros and ones, where the indices of the nonzero elements indicate the location of the leader nodes. L denotes the Laplacian matrix of the system's graph, which is assumed to be connected, and L_h denotes the submatrix of L after all columns and rows corresponding to the leaders (as indicated by the nonzero elements of h) have been removed.

Reference [1] develops the formulation (LS) for a network of n single-integrators $\dot{x}_j = u_j + w_j$, $j = 1,\dots,n$, where x_j, u_j, and w_j respectively denote the state, the control, and the disturbance of the jth node. The nodes are restricted to using only relative information $u_j = -\sum_{i\in\mathcal{N}_j}(x_j - x_i)$, $j = 1,\dots,n$, where $\mathcal{N}_j$ denotes the set of all nodes that communicate with node j. Stacking all variables into vectors yields the state equation of the overall network $\dot{x} = -L\,x + w$, where L is the Laplacian matrix.

If a node is chosen to be a leader then it is assumed to follow the desired trajectory at all times and is unaffected by input disturbances. In the coordinate system considered in [1] (i.e., deviation from desired trajectory) this implies that the state of every leader is identically equal to zero. The latter fact has the effect of eliminating the rows and columns corresponding to leaders from the state equation

Financial support from the National Science Foundation under awards CMMI-0927509 and CMMI-0927720 and under CAREER Award CMMI-06-44793 is gratefully acknowledged.

M. Fardad is with the Department of Electrical Engineering and Computer Science, Syracuse University, NY 13244. F. Lin and M. R. Jovanović are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455. E-mails: makan@syr.edu, fu@umn.edu, mihailo@umn.edu.

$\dot{x} = -L\,x + w$, until only the dynamics of the follower nodes remain

$$\dot{x}_f = -L_h\,x_f + w_f. \qquad \text{(F)}$$

In equation (F), x_f is a vector containing all states corresponding to follower nodes, w_f is the vector of disturbances affecting them, and L_h is the same as in (LS). Finally, [1] considers an $\mathcal{H}_2$ performance objective, which can be shown to be proportional to $\mathrm{trace}\,(L_h^{-1})$.

III. Leader Selection Using Relaxations and Semidefinite Programing

As we next demonstrate, problem (LS) is not convex. In this section, we first use a linear approximation together with a relaxation of the binary constraints to convexify the leader selection problem. We then proceed to develop two different optimization algorithms using the new relaxed formulation. The first method is based on a greedy algorithm, where leaders are chosen one at a time in an iterative procedure. The second method is based on the sparsity-promoting property of the ℓ_1 norm.

A. Convex Relaxations & Approximations

Let H be the diagonal matrix formed from the elements of h,

$$H = \mathrm{diag}\{h\},$$

so that the nonzero diagonal elements of H determine the indices of the leader nodes. Let L_h be the reduced Laplacian defined in Section II. Then

$$\mathrm{trace}\,(L_h^{-1}) = \mathrm{trace}\,((I-H)(G+H\circ L)^{-1}(I-H)), \quad \text{(J)}$$

where $G := (I-H)\,L\,(I-H)$ and $\circ$ denotes elementwise matrix multiplication; see Appendix for a proof. We additionally demonstrate equation (J) with the help of a simple example. Let

$$L = \begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix}, \quad h = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \implies L_h = \begin{bmatrix} d & e \\ e & f \end{bmatrix}.$$

Since

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad I-H = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$G = \begin{bmatrix} 0 & 0 & 0 \\ 0 & d & e \\ 0 & e & f \end{bmatrix}, \quad H\circ L = \begin{bmatrix} a & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

and thus

$$G + H\circ L = \begin{bmatrix} a & 0 \\ 0 & L_h \end{bmatrix},$$

it follows that

$$\mathrm{trace}\,((I-H)\begin{bmatrix} a & 0 \\ 0 & L_h \end{bmatrix}^{-1}(I-H)) = \mathrm{trace}\,(L_h^{-1}).$$

Using (J) and the Schur complement [4], the optimization problem (LS) can be rewritten as

$$\begin{aligned} \text{minimize} \quad & \mathrm{trace}\,(X) \\ \text{subject to} \quad & \begin{bmatrix} X & I-H \\ I-H & G+H\circ L \end{bmatrix} \geq 0 \\ & G = (I-H)\,L\,(I-H) \\ & H_{jj} \in \{0,1\}, \quad j \in 1,\ldots,n \\ & \mathrm{trace}\,(H) = k, \quad H \text{ diagonal} \end{aligned}$$

where $M_1 \geq M_2$ (respectively $M_1 > M_2$) for matrices M_1 and M_2 implies that $M_1 - M_2$ is a positive semidefinite (respectively positive definite) matrix. To demonstrate the equivalence of this problem and (LS), note that whenever $G + H\circ L > 0$,[1]

$$\begin{bmatrix} X & I-H \\ I-H & G+H\circ L \end{bmatrix} \geq 0 \iff X \geq (I-H)(G+H\circ L)^{-1}(I-H).$$

Therefore, to minimize $\mathrm{trace}\,(X)$ subject to the inequality constraint, with H and G fixed, we take $X = (I-H)(G+H\circ L)^{-1}(I-H)$. From (J) it follows that $\mathrm{trace}\,(X) = \mathrm{trace}\,(L_h^{-1})$, which proves equivalence of the new formulation and (LS). For additional details, the reader is referred to [4, Chap. 2].

Setting aside for the moment its combinatorial nature, the optimization problem above is not convex due to the nonlinear way in which the optimization variable H appears in the matrix G. In [5] a linear approximation of the objective function $\mathrm{trace}\,(YZ)$, at the point (Y_0, Z_0), was considered

$$(1/2)\,\mathrm{trace}\,(Y_0Z + YZ_0).$$

The authors use an iterative procedure in which they minimize $\mathrm{trace}\,(Y_0Z + YZ_0)$ for a given feasible pair (Y_0, Z_0), then set $Y_0 \leftarrow Y$, $Z_0 \leftarrow Z$, and repeat. Motivated by this, we redefine the variable G in the objective function (J) as

$$G := (1/2)\,(I-H)\,L\,(I-H_0) + (1/2)\,(I-H_0)\,L\,(I-H),$$

where H_0 is our current-best-estimate of H.

Despite the application of a linear approximation, the optimization problem is still combinatorial and therefore intractable in general. We further apply a relaxation by replacing, for $j \in 1,\ldots,n$, the binary constraints $H_{jj} \in \{0,1\}$ with the linear ones $0 \leq H_{jj} \leq 1$. This is a common procedure in dealing with binary constriants; see for example [6], [7].

The above described linearization in conjunction with a relaxation of the binary constraints in (LS) yields a convex optimization problem which can be written as the semidefi-

[1]The inequality $G + H\circ L > 0$ holds whenever H has at least one nonzero diagonal element. To see this, note that the matrix $(I–H)L(I–H)$, when restricted to the range space of $I-H$, is a principal submatrix of the Laplacian and thus strictly positive. Similarly, the matrix $H\circ L$, when restricted to the range space of H, is a diagonal matrix with strictly positive elements.

nite program (SDP)

$$\begin{aligned} \text{minimize} \quad & \operatorname{trace}(X) \\ \text{subject to} \quad & \begin{bmatrix} X & I-H \\ I-H & G+H\circ L \end{bmatrix} \geq 0 \\ & G = (1/2)(I-H)L(I-H_0) + \\ & \qquad (1/2)(I-H_0)L(I-H) \\ & 0 \leq H \leq I, \quad H \text{ diagonal} \\ & \operatorname{trace}(H) = k \end{aligned} \tag{R}$$

Using the relaxed formulation (R), we proceed in two directions:

1. **Greedy algorithm (Method 1)**: We use the optimization problem to find leaders one at a time. Given κ leaders, we find the best $(\kappa+1)$th leader; we start from zero leaders and continue until all k leaders have been found.
2. **Soft constraint (Method 2)**: We relax the hard constraint $\operatorname{trace}(H) = k$ to a soft one by considering the objective function $\operatorname{trace}(X)+\gamma \operatorname{trace}(H)$. Starting from very large values, we gradually lower γ until H has only k nonzero diagonal elements.

As we demonstrate in Section IV, with the help of an example, both algorithms perform well in the sense that they find solutions with performance values comparable to the globally optimal solution (determined by an exhaustive search).

B. Method 1: Greedy Algorithm

Consider the optimization problem

$$\begin{aligned} \text{minimize} \quad & \operatorname{trace}(X) \\ \text{subject to} \quad & \begin{bmatrix} X & I-H \\ I-H & G+H\circ L \end{bmatrix} \geq 0 \\ & G = (1/2)(I-H)L(I-H_0) + \\ & \qquad (1/2)(I-H_0)L(I-H) \\ & 0 \leq H \leq I, \quad H \text{ diagonal} \\ & \operatorname{trace}(H) = \kappa \\ & H_0 = \operatorname{diag}\{h_0\}, \quad Hh_0 = h_0 \end{aligned} \tag{M1}$$

where the optimization variables are the symmetric matrix X and the diagonal matrix H. This optimization problem is solved as part of an iterative loop (the exact description of the algorithm will be given below). The solution H of every iteration is used to form the current-leader matrix H_0, which is used for linearization in the next iteration. The iterative process is initiated with $\kappa = 0$ leaders, and leaders are chosen one at a time by setting $\kappa \leftarrow \kappa + 1$ in every iteration until all k leaders have been selected. Thus, problem (M1) assumes that $\operatorname{trace}(H_0) = \kappa - 1$; the solution of (M1) will determine the κth leader. The constraint $Hh_0 = h_0$ enforces that H have at least the same nonzero elements on its diagonal as H_0. To see this, observe that H_0 contains the indices for the first $\kappa - 1$ leaders, and in the current iteration we are looking for an H that has the same $\kappa - 1$ leader indices as H_0, plus, the next best leader. Finally, note that the solution H at every step can have values that fall between zero and one, as the binary constraint on the elements of H has been relaxed. We find the index of the largest diagonal element of H that is not in the same location as any of the nonzero elements of H_0; we choose this index as the index of the next leader, and update h_0 accordingly.

In summary, the algorithm can be written as follows.

Greedy Algorithm (Method 1)

given k
Set $\kappa = 0$, $h_0 = 0$, $H_0 = 0$.
while $\kappa < k$ **repeat**

1. $\kappa \leftarrow \kappa + 1$.
2. Solve the semidefinite program (M1).
3. Find the maximum diagonal element of H that does not occur in the same location as any of the nonzero diagonal elements of H_0. If this maximum is zero, stop.
4. Store the diagonal index of the maximum element found in step 3 in max_indx. Set $h_0(\text{max\_indx}) = 1$.
5. Set $H_0 = \operatorname{diag}\{h_0\}$.

end while

In Section IV this algorithm is applied to the 25-agent example considered in [1]; see Fig. 1 for the problem graph. The tables in Figs. 2 and 3 contain results and comparisons of different methods.

C. Method 2: Soft Constraint

Consider the optimization problem

$$\begin{aligned} \text{minimize} \quad & \operatorname{trace}(X) + \gamma \operatorname{trace}(H) \\ \text{subject to} \quad & \begin{bmatrix} X & I-H \\ I-H & G+H\circ L \end{bmatrix} \geq 0 \\ & G = (1/2)(I-H)L(I-H_0) + \\ & \qquad (1/2)(I-H_0)L(I-H) \\ & 0 \leq H \leq I, \quad H \text{ diagonal} \end{aligned} \tag{M2}$$

where the optimization variables are the symmetric matrix X and the diagonal matrix H. As in the greedy algorithm of the previous section, this optimization problem is solved as part of an iterative loop (its exact description is provided below). The solution H of every iteration is treated as the current-best-estimate H_0 for linearization in the next iteration. This method relies on the sparsity-promoting property of the ℓ_1 norm [2], [3]; it is expected that since $\operatorname{trace}(H) = \|h\|_1$ appears in the objective, then the diagonal of H will be sparse. Indeed, it is observed that for general values of γ, the solution of (M2) is in fact sparse, and sparsity increases as γ increases. Note that the solution H at every step can have values that fall between zero and one, as the binary constraint on the elements of H has been relaxed. Starting from very large values, we gradually lower γ until H only has k nonzero diagonal elements. We choose the indices of all nonzero elements of the sparse diagonal matrix H as the indices of the leaders. If it is not possible to find a γ that results in H having exactly k nonzero elements then the leaders are chosen by the indices of the k largest diagonal elements of H.

In summary, the algorithm can be written as follows.

Soft Constraint (Method 2)

given k, γ, ε
1. Set $h_0 = 0$, $H_0 = 0$.
while $\|H - H_0\| > \varepsilon$ **repeat**
Solve the semidefinite program (M2).
Set $H_0 = H$.
end while
2. Set σ equal to the number of nonzero diagonal elements in H. If $\sigma = k$, stop.
3. If $\sigma > k$, increase γ. If $\sigma < k$, decrease γ. Go to step 1.

In Section IV this algorithm is applied to the 25-agent example considered in [1]; see Fig. 1 for the problem graph. The tables in Figs. 2 and 3 contain results and comparisons of different methods.

Remark: The computational complexity of the algorithms described above depends on how efficiently we can solve the SDPs (M1) and (M2). The most efficient SDP solvers (such as SeDuMi) employ interior point methods that require solving systems of linear equations with $O(n^2)$ variables. If problem structure is not utilized, this takes $O(n^6)$ operations. However, for sparse large-scale problems, it is advantageous to utilize iterative schemes [8] (e.g., conjugate gradient method) to *approximately* solve the underlying linear equations. In the *worst-case* scenario, this approach brings down the complexity of solving LMIs to $O(n^{5.5})$ operations, while the *average-case* complexity is close to $O(n^4)$; for additional details, see [9], [10] and references therein. Therefore, the computational complexity of solving (M1) and (M2) can be significantly improved by developing a customized interior point method that exploits the sparsity structures in large-scale leader selection problems. Development of such a method is a topic of our ongoing research.

IV. An Example

In this section we use the example from [1] as a benchmark. We demonstrate that in most cases both of the leader selection algorithms presented here achieve better performance than those developed in [1].

Fig. 1 shows the problem graph, while the tables in Figs. 2 and 3 display the computational results for the different algorithms.

A short description of the columns of the table in Fig. 2 is as follows.

- **Optimal**: gives the globally optimal choice of leaders, as determined in [1] using an exhaustive search applied to the original combinatorial formulation (LS).
- **PatBam 1**: gives results for a greedy algorithm developed in [1] that chooses leaders by determining L_h^{-1} for different vectors h at every iteration.
- **PatBam 2**: gives results for an approximate greedy algorithm developed in [1] that uses upper bounds on eigenvalues of L_h^{-1} to save on computation.
- **Method 1**: gives results for the greedy algorithm described in Sec. III-B.
- **Method 2**: gives results for the soft constraint method described in Sec. III-C. The dagger sign † in this column represents those cases in which it was difficult or impossible to find a γ that resulted in H having exactly k nonzero elements (for example, it turns out that the number of nonzero diagonal elements of H never falls below two no matter how large γ is taken to be); in these cases the leaders were chosen by the indices of the k *largest* diagonal elements of H.

The table in Fig. 3 displays the objective values, $J = \mathrm{trace}\,(L_h^{-1})$, corresponding to each of the leader selections in Fig. 2. Note that for $k = 1$, PatBam 1 computes J for all possible leaders and chooses the leader that results in the smallest J. Hence it is not surprising that PatBam 1 performs well at $k = 1$, as it is computing the global optimum. It is worth mentioning that for Method 2 convergence was always achieved in fewer than ten iterations.

V. Interpretation in Terms of Electrical Networks

In this section, we give an electrical network interpretation to the leader selection problem and the optimization problem (LS) in particular. To this end, we first describe a different way of deriving (LS) that also lends itself well to relaxations and optimization techniques.

Let us assume that the equation

$$\dot{x} \;=\; -(L + aH)\, x \;+\; w$$

governs the dynamics of the entire network, where x is the vector containing the states of all nodes, L is the Laplacian, a is a positive scalar, and $H = \mathrm{diag}\{h\}$ as defined before.

An important observation is that for very large values of a the states of the leader nodes converge to zero very fast. In particular, in the limit of $a \to \infty$ the state of every leader is zero for $t > 0$, which is equivalent to the assumption made in [1]. Furthermore,

$$\mathrm{trace}\,((L + aH)^{-1}) \;\to\; \mathrm{trace}\,(L_h^{-1}) \quad \text{as} \quad a \to \infty.$$

We can thus reformulate the leader selection problem as

$$\begin{aligned} \text{minimize} \quad & \mathrm{trace}\,((L + aH)^{-1}) \\ \text{subject to} \quad & H_{jj} \in \{0, 1\}, \quad j \in 1, \ldots, n \\ & \mathrm{trace}\,(H) = k \end{aligned} \tag{LS'}$$

where a is a large positive number. In a companion paper [11] we further investigate the formulation (LS') and develop efficient methods for computing lower and upper bounds on the global optimal value of the leader selection problem.

It turns out that the Laplacian matrix L, which describes the network's interconnection structure, can also be thought of as the *conductance matrix* of a resistive network [12] in which all edge conductances have unit value. For such an electrical circuit, we have $J = LV$, where V denotes the vector of node potentials and J is the vector of injected currents.

Consider the matrix $Q := L + aH$. Using [13], Q can be associated with an electrical circuit in which the Laplacian L is interpreted as describing links of unit conductance between the nodes of the network, and aH is interpreted as adding *shunt conductances* between the ground and those nodes indicated by the nonzero diagonal elements of H. Fig. 4 (left) gives an example of such a circuit, and Fig. 4 (right) demonstrates its corresponding "loopy Laplacian" matrix [13].

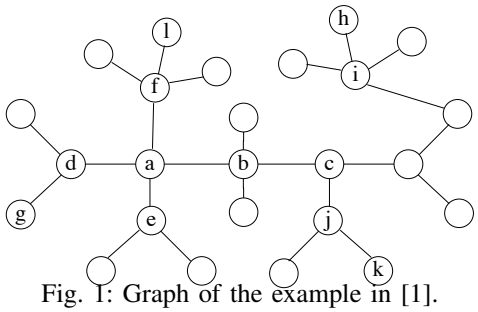

Fig. 1: Graph of the example in [1].

k	Optimal	PatBam 1	PatBam 2	Method 1	Method 2
1	b	b	b	i	a†
2	a,i	b,i	b,h	i,j	a,i
3	a,i,j	b,f,i	b,g,h	f,i,j	a,i,j
4	e,f,i,j	b,f,i,j	b,g,h,l	e,f,i,j	a,f,i,j
5	d,e,f,i,j	b,e,f,i,j	b,g,h,k,l	d,e,f,i,j	d,e,f,i,j†

Fig. 2: Leader choices for different methods applied to the example in [1]. See the table in Fig. 3 for a quantitative comparison of the performance of the leader selections in Fig. 2.

k	Optimal	PatBam 1	PatBam 2	Method 1	Method 2
1	66.00	**66.00**	**66.00**	112.00	69.00
2	38.40	44.75	47.20	64.00	**38.40**
3	30.00	33.25	38.87	32.13	**30.00**
4	25.27	27.36	32.32	**25.27**	26.00
5	20.70	22.19	26.62	**20.70**	**20.70**

Fig. 3: Objective values, $J = \mathrm{trace}\,(L_h^{-1})$, corresponding to each of the leader selections in Fig. 2. The numbers in bold indicate that the value of the objective is equal to the global optimum.

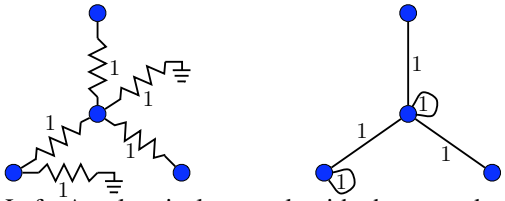

Fig. 4: Left: An electrical network with shunt conductances. Right: The graph of a *loopy Laplacian* [13] associated with an electrical network with shunt conductances; a *self-loop* at node i of the graph corresponds to a shunt conductance connecting node i to the ground.

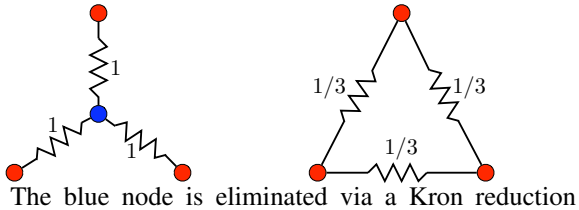

Fig. 5: The blue node is eliminated via a Kron reduction while the red nodes remain [13].

The *effective resistance matrix* R is a matrix whose ijth element R_{ij} is equal to the effective resistance measured across the pair of nodes i and j. The *total effective resistance* R_{tot}, defined as the sum of the effective resistances between all distinct pairs of nodes, or equivalently

$$R_{\mathrm{tot}} \;=\; (1/2)\,\|R\|_{\ell_1} \;=\; (1/2)\sum_{i,j=1}^{n} R_{ij},$$

is an important quantity that arises in a wide range of applications [12]. R_{tot} can be computed directly from the conductance matrix

$$R_{\mathrm{tot}} \;=\; n\,\mathrm{trace}\,(Q^{\dagger}),$$

where † denotes the pseudo-inverse. It can be shown that the total effective resistance is either reduced or remains constant with the addition of shunt conductances between the nodes and the ground.

The concept of total effective resistance gives an electrical network interpretation to the objective function in (LS'). Namely, $\mathrm{trace}\,((L + aH)^{-1})$ is the total effective resistance of the electrical network described by the conductance matrix $Q = L + aH$ in which those nodes indicated by the nonzero elements of H are connected to the ground by shunt resistors with conductance equal to a.

To give an electrical network interpretation to $\mathrm{trace}\,(L_h^{-1})$ we need the concept of *Kron reduction*, which is the reduction of an electrical network through the application of a Schur complement to its associated conductance matrix [13]. Fig. 5 demonstrates an example of Kron reduction.

Consider an electrical network with conductance matrix

Q, and assume that certain nodes $\varphi \subsetneq \{1,\dots,n\}$, $|\varphi| \geq 2$ are to be kept and the rest of the nodes $\lambda = \{1,\dots,n\}\backslash\varphi$ are to be eliminated through a Kron reduction. Then, after an appropriate labeling of the nodes, the conductance matrix can be written as

$$Q = \left[\begin{array}{c|c} Q_{\lambda\lambda} & Q_{\lambda\varphi} \\ \hline Q_{\varphi\lambda} & Q_{\varphi\varphi} \end{array}\right]$$

with its Kron-reduced matrix $Q_{\rm red}$ given by the Schur complement

$$Q_{\rm red} \,=\, Q_{\varphi\varphi} \,-\, Q_{\varphi\lambda}\, Q_{\lambda\lambda}^{-1}\, Q_{\lambda\varphi}.$$

We note that the effective resistance R_{ij} between two nodes $i,j \in \varphi$ is the same regardless of whether it is computed using the original conductance matrix Q or its Kron-reduced version $Q_{\rm red}$ [13]. Now, if we assume that the nodes in the set λ are connected to the ground by resistors with conductance equal to a, then it can be shown that as $a \to \infty$ we have $Q_{\lambda\lambda}^{-1} \to 0$ and thus $Q_{\rm red} \,=\, Q_{\varphi\varphi}$.

The concept of Kron reduction introduced above gives an electrical network interpretation to the objective function in (LS). Namely, $\mathrm{trace}\,(L_h^{-1})$ is the total effective resistance of a circuit in which the nodes indicated by the nonzero elements of h are grounded and eliminated via Kron reduction.

In summary, the optimization problems (LS) and (LS') in Section II can be restated in terms of electrical networks as follows.

- Problem (LS): Choose k nodes to connect to the ground such that the effective resistance of the circuit is minimized.
- Problem (LS'): Choose k nodes to connect to the ground, via shunt conductances of value a, such that the effective resistance of the circuit is minimized.

VI. Conclusions

We consider the problem of devising computationally efficient algorithms for optimal leader selection in networks of dynamical systems. Relaxing the original problem developed in [1] so as to obtain a convex formulation, we propose two algorithms that utilize semidefinite programming to find the optimal leaders.

In a companion paper [11] we examine the leader selection problem for networks in which additive stochastic disturbances enter into the state equations for both leaders and followers. In addition to relative information exchange with their neighbors, leaders also have access to their own states. This setup may be encountered in multi-vehicle coordination problems where a certain number of agents (leaders) may be equipped with GPS devices. For networks of single-integrators, we show that this formulation leads to an objective function that is convex, thereby avoiding the need for its linearization. Furthermore, we develop efficient methods to compute lower and upper bounds on the global optimal value of the leader selection problem. In particular, we consider a convex relaxation to obtain a lower bound and we use simple but efficient greedy algorithms to obtain an upper bound. Finally, we employ the alternating direction method of multipliers to search for a local minimum in the presence of non-convex Boolean constraints.

Appendix

Proof of equation (J): Let h be the vector that determines the leader nodes, i.e., a vector composed of zeros and ones, where the indices of the k nonzero elements indicate the location of the k leaders. After an appropriate relabeling of the nodes, the vector h and the Laplacian L can be written as (for ease of notation, we use the same letters to denote h and L before and after the relabeling)

$$h \,=\, \left[\begin{array}{c} \mathbb{1} \\ 0 \end{array}\right], \quad L \,=\, \left[\begin{array}{cc} L_l & L_0 \\ L_0^T & L_h \end{array}\right],$$

where $\mathbb{1}$ is a $k\times 1$ vector of all ones, and L_l, L_0, and L_h are matrices of dimensions $k\times k$, $k\times(n-k)$, and $(n-k)\times(n-k)$, respectively. Let $p := n-k$. Then from the definitions $H = \mathrm{diag}\{h\}$ and $G = (I-H)\,L\,(I-H)$ it follows that

$$H \,=\, \left[\begin{array}{cc} I_{k\times k} & 0_{k\times p} \\ 0_{p\times k} & 0_{p\times p} \end{array}\right], \quad I-H \,=\, \left[\begin{array}{cc} 0_{k\times k} & 0_{k\times p} \\ 0_{p\times k} & I_{p\times p} \end{array}\right],$$

$$G \,=\, \left[\begin{array}{cc} 0_{k\times k} & 0_{k\times p} \\ 0_{p\times k} & L_h \end{array}\right], \quad H\circ L \,=\, \left[\begin{array}{cc} I_{k\times k}\circ L_l & 0_{k\times p} \\ 0_{p\times k} & 0_{p\times p} \end{array}\right].$$

Thus

$$G + H\circ L \,=\, \left[\begin{array}{cc} I_{k\times k}\circ L_l & 0_{k\times p} \\ 0_{p\times k} & L_h \end{array}\right],$$

and we have

$$\mathrm{trace}\,\Big((I-H)\left[\begin{array}{cc} I_{k\times k}\circ L_l & 0_{k\times p} \\ 0_{p\times k} & L_h \end{array}\right]^{-1}(I-H)\Big) \,=\, \mathrm{trace}\,(L_h^{-1}),$$

which is the desired result.

References

[1] S. Patterson and B. Bamieh, "Leader selection for optimal network coherence," in *Proceedings of the 49th IEEE Conference on Decision and Control*, 2010, pp. 2692–2697.

[2] E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[3] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, pp. 877–905, 2008.

[4] S. Boyd and L. El Ghaoui, *Linear Matrix Inequalities in System and Control Theory*. SIAM, 1994.

[5] L. El Ghaoui, F. Oustry, and M. AitRami, "A cone complementarity linearization algorithm for static output-feedback and related problems," *IEEE Trans. Automat. Control*, vol. 42, no. 8, pp. 1171–1176, 1997.

[6] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Processing*, vol. 57, no. 2, pp. 451–462, 2009.

[7] A. Ghosh and S. Boyd, "Growing well-connected graphs," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006, pp. 6605–6611.

[8] C. C. Paige and M. A. Saunders, "LSQR: An algorithm for sparse linear equations and sparse least squares," *ACM Transactions on Mathematical Software*, vol. 8, no. 1, pp. 43–71, 1982.

[9] L. Vandenberghe and S. Boyd, "A primal-dual potential reduction method for problems involving matrix inequalities," *Mathematical Programming*, vol. 69, no. 1, pp. 205–236, 1995.

[10] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Review*, vol. 38, no. 1, pp. 49–95, 1996.

[11] F. Lin, M. Fardad, and M. R. Jovanović, "Algorithms for leader selection in large dynamical networks: Noise-corrupted leaders," in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, 2011, to appear.

[12] A. Ghosh, S. Boyd, and A. Saberi, "Minimizing the effective resistance of a graph," *SIAM Review*, vol. 50, no. 1, pp. 37–66, 2008.

[13] F. Dorfler and F. Bullo, "Kron reduction of graphs with applications to electrical networks," *SIAM Review*, 2011, submitted.